# A Concurrent Design Approach for Designing Space Telescopes and Instruments

Knut I. Oxnevad

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109-8099

## ABSTRACT

At the Project Design Center at the Jet Propulsion Laboratory, a concurrent design environment is under development for supporting development and analyses of space instruments and telescopes in the early, conceptual design phases. This environment is being utilized by a Team I, a multidisciplinary group of experts. Team I is providing study and proposal support. To provide the required support, the Team I concurrent design environment features effectively interconnected high-end optics, CAD, and thermal design and analysis tools. Innovative approaches for linking tools, and for transferring files between applications have been implemented. These approaches together with effective sharing of geometry between the optics, CAD, and thermal tools are already showing significant timesavings.

**Keywords:** Concurrent Engineering, Graphical Programming, Multidisciplinary Teams, Conceptual Design, Space Telescopes and Instruments, Total Systems Approach

## 1. INTRODUCTION

At the Jet Propulsion Laboratory (JPL), large resources are put into efforts aiming at improving and changing the organization to effectively deal with developing smaller missions in the hundred million, rather than in the billion dollar range. A large number of these missions are won based on competitive proposals in response to Announcement of Opportunities (AO's) from NASA headquarters. Writing and developing proposals is, therefore, becoming increasingly important for JPL.

To support the high number of proposals, this year about 50, a proposal support infrastructure has been developed. The Project Design Center (PDC) which has been developed over some 3 years forms the core of this support. The initial aim[1,2] of the PDC was to support various projects and proposals with early conceptual mission design analysis capabilities. These conceptual mission designs are developed in a couple of weeks by a multidisciplinary team of specialists that work concurrently, in the same room, together with their customers on a set of computers (stations) and tools that are linked. The team, called team X, utilizes Macintosh computers, Excel spreadsheets, and publish and subscribe for linking these spreadsheets together. This provides a very powerful environment for conceptual mission design studies.

In late 1996, it was decided that there was also a need for a team that could provide early conceptual design analysis support for instrument development and instrument proposal work. This led to the development and implementation of Team I. Team I builds on the same general principles as Team X, but different customer requirements led to a different implementation. Typically, an instrument proposal requires a higher degree of detail in both the optical, radiometric, and thermal analyses. Team I is, therefore, utilizing what is considered high end tools, rather than spreadsheets for their development and analysis work.

The initial version of Team I was set up to primarily support optical instrument work. The Team I set of tools, consequently, includes tools such as Code V, ZeMax, TracePro, Mechanical Desk Top (MDT), and RadCAD/SINDA. Most of the Team I tools are running on PC NT platforms. The plan is to modify the Team I composition and tool set to also support other types of instruments.

Integrating these high end tools in a user friendly way, and making the passing of data between them effective and happen in near real time required the Team I development group, headed by the author, to employ some innovative approaches[3]. The development of executives, utilizing a graphical programming language, represents one such approach. This language makes code generation as well as code modification relatively easy. The latter is especially important in an environment requiring rapid adaptation to new customer needs.

Team I includes experts from the areas of optical analysis and design, mechanical and CAD design, thermal analysis, costing, radiometry, and electronics analysis and design. Each of these experts man a computer with the appropriate tools. During the development phase these experts worked very closely with the development group. These experts are also playing an active role together with Team I customers in the continuous development and improvements taking place as more operational experience is being gained.

This paper starts with a description of the Team I environment and ends with a discussion about some of the approaches utilized in the Team I environment that are showing great potential for time savings. The Team I environment is described in section 2, its stations in section 2.1, its tools in section 2.2, and its process in section 2.3. Team I unique approaches for saving time are discussed in section 3. Some of these approaches are the computerizing of preliminary calculations, section 3.1; the automating of data transfers between tools, section 3.2; and the sharing of common geometry between applications, section 3.3.

## 2. THE TEAM I ENVIRONMENT

### 2.1. The Stations

In its current configuration, Team I includes a mechanical/CAD station, an optics station, a radiometry station, a thermal station, a cost station, and a system station. There are plans for incorporating an electronics station and a documentation station into the mix.

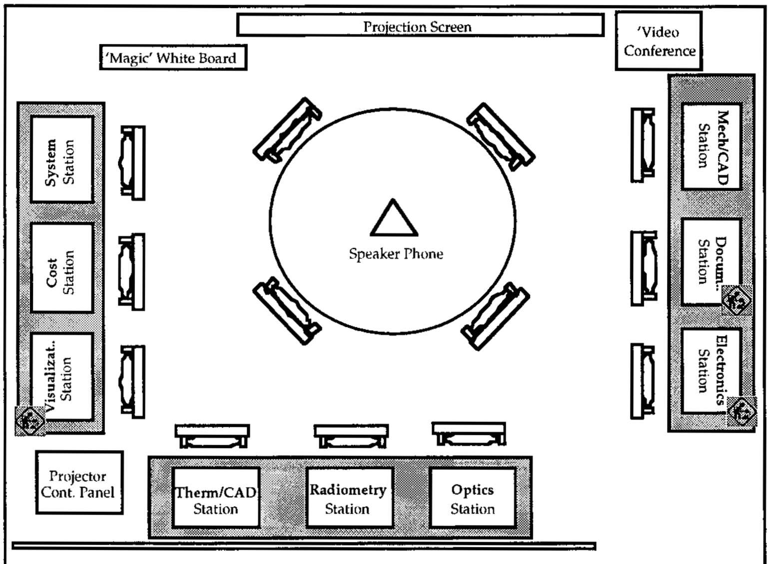

Figure 1: The Team I Room

The current Team I configuration is primarily set up to support the design and analysis of optical instruments. However, the Team I environment was made to be flexible and be able to change with changing customer needs. Changes may effect both the station and tools mix. The document station and the electronics station are currently not active, and they will, therefore, not be discussed any further. The visualization station, also called the orbital analysis station, is for now incorporated into the system station and will briefly be discussed under that station. To improve group interactions, any station's display can be shown on the large projection screen in front of the Team I room, shown in Figure 1: The Team I Room, on this page.

At the **optics station,** an optical designer and analyst uses variables such as number of wavelengths, aperture diameter, F#, field angle (degrees), temperature, pressure, mirror/lens surface types, and type of mirror material for designing the right optics configuration. The tools Code V and ZeMax are used for this part of the design and analysis work. The geometric representation of the surfaces of the selected optics configuration, together with the geometric representation of the resulting rays, are provided as an IGES file. Additionally, the optics configurations itself can be ported to TracePro (ACIS based), also on the optics station, and turned into ACIS based solids and provided as SAT files.

These SAT files can be exchanged between any ACIS based programs. Mechanical Desktop (MDT) is one such program. Cost and mass estimates of the developed optics configuration can also be provided. The ACIS engine is developed by Spatial Technology.

At the **radiometry station**, variables such as required temperature, quantum efficiency, dark current level and readout noise of the detectors, #bits/pixel, aperture diameter, F#, spectral resolution, target scene reflectivity, altitude, number of bands, and observed wavelengths are used for ensuring that minimum signal to raise (S/N) ratios are achieved, and for calculating noise equivalent temperature (NEΔT) curves. The tools used for these calculations was developed by the radiometry analyst in Excel spreadsheets.

The mechanical designer sits at the **mechanical/CAD station**. His/her job is to design support structures (holders for lenses, mirrors, and detectors) and enclosures around the optics configuration provided as output from the optics station. The geometry of these set ups is provided in IGES and SAT formats. MDT, which is also built on the ACIS engine is currently being used as the Team I CAD tool. The station also provides preliminary mass, volume, and area estimates of the developed instrument design. The mechanical/CAD work can be time-consuming, and it is, therefore, mostly done before or after a concurrent session. More about this in section 2.3, The Process, p. 5.

At the **thermal station,** a combination of RadCAD and SINDA tools are used. RadCAD uses the geometry developed on the mechanical/CAD station together with orbital parameters, for calculating orbital heating rates, and for producing radiation interchange factors. SINDA, a thermal analysis program, utilizes these results for calculating detector and optics temperatures for the given orbital environments. This information is then used for discussions about radiator placing, and about whether active or passive cooling is required.

The **cost station** is manned by a cost expert that will perform either grassroots costing (costing by analogy) or parametric costing. The parametric cost models take into account factors such as mass, type of technology, development time, and complexity of instrument part. Output from the cost station is fed into the system station.

At the **system station**, the high level mission parameters (inputs) are defined at the beginning of the session. The main output variables are also sent to and displayed on this station. Some of the high level mission parameters are type of mission; type of orbit; the classical orbital parameters, semi major axis, orbital inclination (calculated for Sun synchronous orbits), right ascension of ascending node, argument of perigee, true anomaly, and observation time and date; orbital time (calculated), orbital velocity calculated (rad/s, and km/s); orbiting body (Earth, Mars, etc.); surface temperature, reflectivity of orbiting body; wavelengths to be observed at; and number of bands. The main output variables are instrument mass and cost, and the power required by the instrument. Preliminary estimations of instrument datarates and communication downlink data rates will also be calculated and displayed on the system station. The system station was put in place primarily to ensure that all applications would be using the same high level system parameters at all points in the design cycle. This is achieved by the system station making these high level parameters available to the various Team I applications in a format that they can read. In the same way, data from the various applications are extracted from their output files and displayed on the system station. This work is under development. LabVIEW has been used for developing file data extraction routines, file building routines, and routines for exchanging data between the team I applications and the system station. Collectively these routines or programs are called executives. The Team I orbital analysis and display software (SOAP) are also located on the system station, so is the electronic white board used in many Team I sessions.

## 2.2. The Tools

**Mechanical Desktop**[4] **(MDT)** is a parametric, ACIS 3.0 and feature-based 3-D solid modeling engine. The program supports blended NURBS surface features, and nonconstrained and constrained design approaches. Dimensions and annotations are automatically displayed in drawing view. The program can be used for calculating surface area, mass, moments of inertia, and volume of parts and assemblies. Files can be imported and exported utilizing fileformats such as IGES Version 5.3, STL (stereolithography), DWF, DWG, DXF, IDF, VRML, SAT (ACIS), 3DS (3D Studio®), WMF (Windows Metafile), EPS (Encapsulated Post Script), and BMP. MDT supports the AutoLISP® programming language, and application programming interfaces (API's) provide C language access to MDT features, functions, and attributes.

**CODE V®**[5] is a non sequential optical design, analysis, illumination calculations, and fabrication support program, that features optimization, various analyses types, and tolerancing.

Optimization routines include , transverse ray errors for pre-stored ray patterns, wavefront variance for pre-stored ray patterns, modulation transfer function (MTF) at user-selected spatial frequencies, and Global Synthesis®.

Supported analysis capabilities include, geometric performance analysis, diffraction analysis, illumination analysis, spectral analysis, and environmental analysis. In the geometric performance analysis, the program utilizes, third-order aberration analysis, real ray tracing, MTF and square-wave responses, radial energy distribution, line spread distributions, detector energy analysis, scanned quadrant detector analysis, cat's eye diagrams, and footprint analyses. Diffraction analysis is provided through root mean square (RMS) wavefront error and Strehl definition, the polychromatic point spread function, encircled energy, and MTF. The illumination analysis features Monte Carlo simulation, and includes analysis of undesired radiation, location and evaluation of ghost images, narcissus calculation for calculating equivalent differential temperature of integrated surface/detector retro-reflections. In the spectral analysis, based on a specified detector, black body, or filter, the system spectral response curve is calculated together with appropriate sampling wavelengths and weights for polychromatic computations. In the environmental analysis, changes in ambient temperature and pressure in the optical system, including expansion of lens spacers, are simulated.

Semi-automatic tolerancing is provided based on RMS wavefront error, diffraction MTF, or chief ray distortion. The program also has capabilities for calculating system weight and center of mass, and it supports the export of files in the IGES format.

**ZeMax**[6] is a non sequential optical design analysis program that features optimization routines, various analyses types, and tolerancing.

Optimization routines include actively damped least squares, minimization of peak-to-valley or RMS of either spot radius, x, y, x+y, or wavefront error, referenced to either the chief ray or the centroid, and MTF response and encircled energy. Equality, inequality, and Lagrange multiplier constraints are all supported with arbitrary weighting. The program can optimize most of the parameters in a optical system, including radii, thicknesses, glasses, conics, aspheric coefficients, grating spacings, apertures, wavelengths, and fields.

Analysis types include spot diagram analysis, diffraction analysis, and encircled energy analysis. Spot diagrams include, standard field-by-field, through focus, full field, matrix (spot diagrams separated by field and wavelength). Diffraction analysis functions supported include, MTF, square wave MTF, through focus MTF (sine or square), point spread function, point spread function cross section, surface 3D MTF, geometric transfer function (GTF), and through focus GTF. As for the encircled energy analyses, diffraction radial, geometric radial, x, y, and line and edge response analyses are supported. The program also features analysis of grid distortion, longitudinal aberration, lateral color field curvature and distortion, RMS vs. field, RMS vs. focus, extended source imaging capability, interferograms, chromatic focal shift, glass internal transmittance vs. wavelength, and relative illumination.

The program utilizes both sensitivity analysis, and Monte Carlo simulations for performing tolerancing, and RMS spot size, RMS wavefront error, MTF, as well as user defined criteria are supported. Tolerancing can be performed on radii, thickness, lens position, tilt, decenter, irregularity, and wedge. Inverse tolerances, giving maximum acceptable increase in performance are also given. The IGES export file format is supported.

**TracePro**[7] accounts for absorption, specular reflection and refraction, scattering, and aperture diffraction of light propagating through a solid model. TracePro can be used for stray light analysis, non-imaging analysis, radiometry analysis, or any application where the accurate determination of distributions and intensities of light is required. The program includes features for importing lens design data from most lens design applications for creating accurate solid models of lenses with no loss of information. Being based on the ACIS geometric modeling engine, the program can effectively share solid model data with other ACIS based applications. For non ACIS based CAD and analysis programs, data can be exchanged via IGES and STEP files.

**RadCAD**[8] is PC based radiation analysis system that combines the analysis capabilities of programs such as TRASYS with CAD techniques and integrates CAD generated surfaces with parameter-based (TRASYS-like) surfaces. The program works within MDT. From a MDT supplied geometric description of a set of surfaces along with optical property data or from a TRASYS input file, the program can produce form factors and radiation conductors ("RADKs") ready for import into SINDA. The program supports the TRASYS ORBIT1 or ORBIT2 conventions, defines and displays the S/C orbit and orientation, and calculates the orbital heating rates. Support for the creation and manipulation of optical property databases is also provided. CAD 3D faces, regular MxN meshes, and arbitrary polyface meshes are supported by the program.

**SINDA**[9] is a user-extensible finite-difference, lumped parameter (circuit or network analogy) software tool for analyzing complex thermal systems. The program deals with radiation, conduction, and convection, and it includes system performance simulation capabilities, steady state and transient solution capabilities, goal seeking capabilities, optimization capabilities (multiple variables), time and temperature-varying properties, user-determined solution techniques, solution sequences, accuracy levels, and outputs, and spreadsheet-like expressions and user variables (registers).

**LabVIEW**[10] is a cross-platform (UNIX, PC, Mac, and PowerPC) program development tool similar to, for example, C. It uses a graphical programming language called G rather than a text-based programming language for creating code. A LabVIEW program will appear in a block diagram form. The boxes in the block diagram can represent user defined programs, subroutines, or subsystems, or built in functions and subroutines. These boxes are recognized through their icons. For user-defined subroutines, the icons are drawn by the user. Functions or subroutines are, in the LabVIEW vocabulary, called virtual instruments (VI's), because they imitate the appearance and operations of an actual instrument. A VI includes a front panel and a (block) diagram. The front panel is the VI's interactive user interface. In the front panel, the model developer will define all input (controls) and output (indicator) variables. The relationships between these input and output variables are defined in the block diagram.

## 2.3. The Process

The Team I process is under development, and the process that is being described here is what the Team I development group is working towards. The process, which is shown in Figure 2: The Design Process, p. 5, starts with the Team I sending a potential customer a list of input variables that the team needs for setting up for a Team I session. This list is currently being developed by the Team I development group with input from all Team I members. About a week after the list has been sent to the customer, Team I and the customer meet to discuss the input list and plan the Team I sessions. At this meeting, the customer will also be briefed on the workings of Team I. Based on the input from this meeting, the Team I members and developers will prepare the session. These preparations will vary in volume and complexity depending on the type of support the customer requires, complexity of instrument, and inheritance from other instruments.

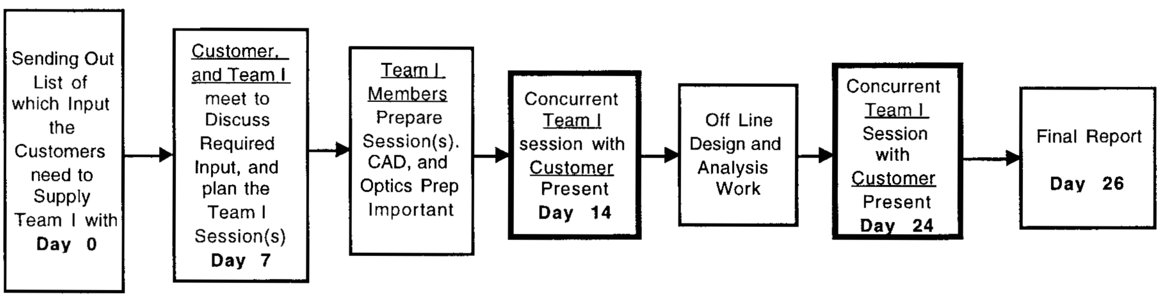


Figure 2: The Design Process

The radiometry expert may have to modify his models to make them work with the specified number of bands, the spectrum being analyzed, and the number of detectors required. The optics expert will have to do preliminary design and analysis on an optics (lenses, or mirrors) configuration that meets customer requirements on orbital altitude, ground resolution (instantaneous field of view: IFOV), swath width, and aperture size. This work easily takes a week. The optics expert will use Code V or ZeMax for this work. At the mechanical/CAD station, a preliminary model of the instrument will be put together. This will be used for visualizing the design and helping the Team I members better understand issues, such as orientation towards the sun, direction of the orbital velocity vector, and direction to the orbiting object. On the system station, the main input parameters will be set up, and calculations of variables such as inclination, orbital time, and orbital velocity will be performed. Possible modifications to the system front page will also be performed to meet special requests from the customer. The orbital parameters from the system station will be used for setting up SOAP for visualizing and analyzing the selected orbit. The thermal expert will also utilize these parameters for setting up RadCAD.

The aim of all of these preparations, is to have the tools prepared to a point where, it becomes possible to perform changes and trades, in a concurrent fashion.

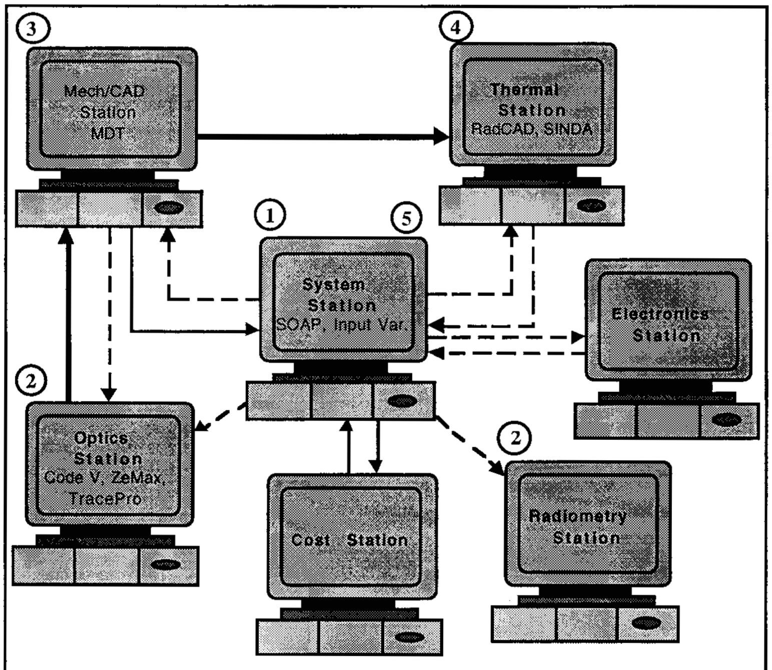

Figure 3: Interconnected, High-End Analysis, and Design Tools

On about day 14, the customer and the Team I members meet to conduct a concurrent design session. The main flows and interactions between the Team I stations are illustrated in Figure 3: Interconnected, High-End Analysis, and Design Tools, p. 6. These sessions are lead by the Team I leader.

Encircled numbers (1...5) show the main steps in the design process. These are discussed below. Dotted lines indicate that electronic transfer of data between applications on the shown stations is under development or under consideration for development. Complete lines indicate that electronic data currently can be transferred and used by applications on the station at the end of the arrow. Connections being regarded as important, especially from a timesavings perspective, are marked with thick complete lines. These connections are discussed in more detail in section 3.3, Sharing Common Geometry, p. 8.

The Team I session would start (1) with the Team, together, furnishing the high level mission parameters for the system station. At this point the system station display would be shown on the large projection screen. Ref. Figure 1: The Team I Room, p. 2. Having the team do this together, helps getting it on the same page from the very beginning of the session. The system station serves both as a placeholder, as well as a point of distribution for these common high level parameters. Preliminary calculations of orbital time, velocity, inclination are also performed. Next, for visualizing the mission, the six orbital parameters are fed automatically through a routine into a SOAP .orb file. This file is then automatically opened and displayed on the system station and on the large projection screen. As the dotted arrows in the figure above indicate, there also are plans for electronically transferring high level parameters to other applications. After the high level mission parameters are established, typically work starts in parallel, and interactively, on the optics station, and on the radiometry station (2). Aperture diameter, detector placing, and focal length, are defined on the radiometry station, and utilized by the optics station, using Code V, or ZeMax, for setting up, and analyzing initial, and prepared optics configurations. After some back and forth between these stations, a recommended optics configuration should emerge. The geometric representation of this configuration is transferred to TracePro and turned into ACIS solids. These are saved in a SAT file format. In parallel, the same geometry plus the resulting rays are saved as an IGES file. The two files are read into MDT (3), on the mechanical/CAD station. Next, the geometries in these two files are aligned, ensuring that the surfaces from the IGES file, exactly match the corresponding surfaces of the TracePro developed solids lenses, mirrors, and detectors. This process also perfectly aligns the rays. At the mech/CAD station, support structure, an enclosure, radiators, and electronics are added. Some of these parts may have been prepared before the session or during the time the time of the optics analysis. As part of the Team I development effort, we are looking into effective ways of parameterizing key dimensions of the support structures, making it possible to modify and use them for a variety of mirror and lens shapes, and sizes. The final CAD drawing will in most cases be finished after the concurrent session. However, after an initial geometry, including lenses, and support structures has been developed, it will be used directly by RadCAD (4) for calculating orbital heating rates and for producing radiation conductors. For this purpose, a very detailed geometric representation of the instrument is not required. Prior to getting the geometry from the mech/CAD station, preliminary orbital heating rates can, therefore, be calculated based on the simple geometry of a box dimensioned as the final instrument and the orbital parameters provided from the system station. From the RadCAD analysis, a file containing form factors and radiation conductors is imported to SINDA for calculating detector and optics temperatures. The RadCAD to SINDA connection is still under development, and whether the SINDA analysis can be performed in a concurrent session or not has yet to be determined. The MDT developed solids geometry (lenses, support structures, and detectors) may be saved as a SAT file, and transferred back to TracePro for stray light analyses.

After all these analyses have been performed, and there is agreement that the resulting design meets the set design criteria, mass and cost will be calculated and displayed on the system station (5) together with other main output variables such as data rate and power. Datarate calculations will initially be performed on the system station. Mass will be calculated in two ways, one based on volume and densities as defined in MDT and Code V (lenses and mirrors), and one based on mass relationships derived from expert knowledge from the Team I members. For mass estimates provided by MDT, a routine has been developed for automatically extracting this information from the MDT mass file, and displaying it on the system station. The instrument costing process is still under development.

Most likely, issues will arise during this first session that may require more detailed analysis that should be done off line. A likely scenario is, therefore, that after the first concurrent session the team and the customer take a break for about 10 days. This should give the customer time to consider the initial results, and the team members time to do required off-line design and analysis work. Any changes coming from the customer should be conveyed to the team members in good time for them to prepare for the second concurrent session at day 24.

The design and analysis process for this session will be similar to that of the first session, with the exception that a report has to be generated at the end of it. Currently, this report is generated jointly by the Team I members during the session. The final report will be delivered to the customer a couple of days later. If no design has been reached that satisfies the customer requirements or if the instrument is too complex for just two concurrent sessions, the report will be postponed to the last concurrent session between the customer and Team I.

## 3. REDUCING DEVELOPMENT AND ANALYSIS TIME

### 3.1. Computerizing Preliminary Calculations

After having gone through some concurrent sessions, utilizing parts of the Team I process, it seems that time can be saved by reducing the number of reoccurring manual calculations done on pieces of paper at the beginning of these sessions. Typically, these calculations range from orbital time, to estimates of aperture diameter. It was assumed that time could be saved by making these calculations and their results electronically available.

The Team I development group, therefore, is looking into using G-code developed by National Instruments, for developing small programs, above called executives, that will perform these calculations. These programs or virtual instruments (VI) will be linked directly to the high level parameters on the system station, also developed in G-code. This will make it possible to perform effective and numerous trade-off analysis based on those parameters. Both input and output variable values from those trade-off analyses will be stored in a file for later retrieval and analysis. This file is set up on the system station and termed the history file. The calculations to be performed in this way are primarily mission related and high level calculations, not performed on any of the other stations. So far, VI's for calculating orbital time, velocity, inclination of Sun synchronous orbits, aperture diameter, F#, have been implemented in the Team I environment. More VI's will be added. The benefit of using G-code is that it is simple to learn, easy to develop in, and easy to modify. These features enables rapid development and modifications of VI's to make them fit any new customer's demands. Additionally, these features make it possible to involve the system experts in the model (VI) development process,[11,12] The approach and methodology utilized in this work was developed and demonstrated in earlier research by the author.[13]

### 3.2. Automating the Transfer of Data between Applications

Very early in the Team I development effort it was recognized that to have high-end tools with unique file formats work together in a near real-time fashion, there was a need for developing effective ways of transferring data between them. Without this capability, too much time would be spent on duplicating manual data entries, and manually searching through and possibly modifying input and output files.

The Team I development group solved this problem by developing routines, (VI's) in G-code, for extracting information from, as well as for inserting code and values into any file type by the press of a button. For example, a routine has been built for automatically extracting mass data from a summary mass file from MDT, transferring it to, and displaying it on the system station. An associated routine transfers this mass data from a system station file into assigned cells in the parametric cost model spreadsheet. Another routine has been developed for inserting orbital elements and other variables from the system station into a unique SOAP formatted orb. file. Through the push of a button on the system station, SOAP opens up this file and displays visually the orbit of the instrument, its ground track, and any other parameters defined from the system station. The latter routine shows how G-code can be used for opening applications and files, a capability that will be useful for performing semi-automatic trade studies, with people in the loop, across a number of tools. These types of trade studies will be looked into in later phases of the Team I development effort.

### 3.3. Sharing Common Geometry

The largest time savings seem to come as a result of the optics, the CAD, and the thermal tools being able to utilize the same geometric data. These time savings can be attributed to progress made by the developers of these tools, and the mix and the utilization these tools in the Team I environment. In many ways, the requirements of Team I forced an effective integration of the optics, CAD, and thermal tools.

According to mechanical designers who have been designing support structures, fasteners, etc. for space instruments for a number of years, the Team I approach, typically, saves them 7-10 days worth of work. Up to recently, they had to manually translate the dimensions and positions (x, y, z) of lenses, mirrors, and detectors from, for example, a Code V output file, into a 2D CAD drawing. Often, such output files would be provided as print outs. In the Team I environment, these steps are eliminated, and the geometry of the optics configuration (dimensions and positions) is transferred automatically without any human intervention, except for opening, and saving files, from the optics tools to the CAD tools. The details of how this is done is discussed in section 2.3, The Process, p. 5.

On the thermal side, the situation was similar. Dimensions and positions (x,y,z) of surfaces had to manually be read from a CAD drawing and then typed into, for example, TRASYS or NEVADA formatted ASCII files. This procedure required that the analyst knew how to set up these fileformats, and that he/she also knew which commands to use for inserting dimension and position values. By using RadCAD instead, CAD geometry from MDT is utilized directly in the RadCAD environment. RadCAD, which works within MDT, provides a visual representation of the geometry, and surfaces to be analyzed, and provides the analyst with a graphical user interface. These features, according to thermal experts, may reduce the time for calculating, radiation interchange factors and orbital heat inputs by a factor of 5.

## 4. CONCLUSIONS

The Team I development is very much at its early stages. However, both team members and customers are starting to see the benefits of utilizing this concurrent and multidisciplinary design and analysis environment. Much good work has been done in interconnecting and making the Team I tools effectively work together. There is still much work to do in this area, especially in transferring high level system parameters to the various applications. More work is also needed in developing high level preliminary analyses capabilities. Later, as more operational experience is gained, the Team I concept will be expanded to include design and analysis capabilities for non optical space instruments, and probably also in-situ instruments. From this perspective, the Team I environment can be seen as a laboratory for developing effective early conceptual design environments for demanding types of space instruments.

## 5. ACKNOWLEDGMENTS

During the Team I development effort, the author has benefited from discussions with numerous people at JPL, especially, listed alphabetically, Apostolis Deslis, Edward C. Hagerott, Randall C. Hein, Pantazis Mouroulis, David D. Norris, Stanley W. Petrick, James F. Schmidling, George Siebes, Jeffrey L. Smith, Harold R. Sobel, and Fred Vescelus.

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

## 6. REFERENCES

1. Casani, K. Thomas, N., *FST - Faster, Better, Cheaper*, SPIE-94, Conference.
2. Project Design Center, Jet Propulsion Laboratory (JPL), Brochure JPL 400-533, July 1994.
3. Oxnevad, K. I., *A Total Systems Analysis Method, for The Conceptual Design of Spacecraft: An Application to Remote Sensing Imager Systems*, Old Dominion University, Ph.D. Dissertation, pp. 13-22, May 1996.
4. AutoDesk, WWW: http://www.autodesk.com/prodsol/mech/products/desktop/techspec/index.htm.
5. Optical Research Associates, WWW: http://www.opticalres.com/cv/cvtechds.html, 1998.
6. Focus Software, WWW: http://www.focus-software.com/zemax/, 1998
7. Lambda Research, WWW: http://www.lambdares.com/PRODUCTS/products.html, 1998.
8. Cullimore and Ring Technologies, WWW: http://www.webcom.com/crtech/radcad.html, 1998.
9. Cullimore and Ring Technologies, WWW: http://www.webcom.com/crtech/sfcap.html, 1998.
10. LabVIEW for Macintosh, User Manual, National Instruments, Part Number 320591-01, December 1993 Edition.
11. Wise, R.M., "LabVIEW our Choice for Cardiothoracic Research", Instrumentation Newsletter, Technical News from National Instruments, Vol. 7, p. A-5, Autumn 1995.
12. Clarkson, M., "Tools on the Fly: Virtual Instruments Take Off", Desktop Engineering, pp. 20-28, September/October 1995.
13. Oxnevad, K. I., pp. 1-96.